

**CLAIMS:**

What is claimed is:

1. A method for modifying computer program instructions  
5 during execution of those instructions, the method  
comprising computer-implemented steps of:  
writing a first value into a memory location,  
wherein the first value represents a first instruction,  
and wherein the first instruction is a patch class  
10 instruction;  
fetching the first instruction from the memory  
location;  
executing the first instruction; and  
overwriting a second value into the memory location,  
15 wherein the second value represents a second instruction,  
and wherein the second instruction is a patch class  
instruction;  
wherein the overwriting may be concurrent with the  
execution of the first instruction.  
20
2. The method according to claim 1, further comprising:  
refetching the first instruction from the memory  
location; and  
reexecuting the first instruction  
25
3. The method according to claim 1, further comprising:  
fetching the second instruction from the memory  
location; and  
executing the second instruction.  
30
4. The method according to claim 1, further comprising:  
reconciling a processor's execution pipeline with

05918313.073101  
TOTAL 13 PAGES

5. The method according to claim 1, wherein the patch class instructions include no-operation instructions and branch instructions.

7. The method according to claim 1, wherein the steps are implemented in a uniprocessor computer system.

8. A computer program product in a computer readable medium for use in a data processing system, for modifying computer program instructions during execution of those instructions, the computer program product comprising:

instructions for overwriting a second value into the memory location, wherein the second value represents a second instruction, and wherein the second instruction is a patch class instruction;

30

9. The computer program product according to claim 8, further comprising:

instructions for reconciling a processor's execution pipeline with the memory location, wherein the reconciliation ensures that the second instruction will be fetched and executed from the memory location if the program subsequently returns to that memory location.

10. The computer program product according to claim 8, wherein the patch class instructions include no-operation instructions and branch instructions.

10

11. The computer program product according to claim 8, wherein the steps are implemented by a multithreaded program.

15 12. The computer program product according to claim 8, wherein the steps are implemented by a single-threaded program.

13. A system for modifying computer program instructions during execution of those instructions, the system comprising:

20 a writing component which writes a first value into a memory location, wherein the first value represents a first instruction, and wherein the first instruction is a patch class instruction;

25 a fetching component which fetches the first instruction from the memory location;

a processing component which executes the first instruction; and

30 an overwriting component which overwrites a second value into the memory location, wherein the second value

404870-070

Docket No. AUS920010676US1

represents a second instruction, and wherein the second instruction is a patch class instruction;

wherein the overwriting may be concurrent with the execution of the first instruction.

5

14. The system according to claim 13, further comprising:

a refetching component which refetches the first instruction from the memory location; and

10 a processing component which reexecutes the first instruction.

15. The system according to claim 13, further comprising:

15 a fetching component which fetches the second instruction from the memory location; and

a processing component which executes the second instruction.

20 16. The system according to claim 13, further comprising:

a reconciliation component which reconciles a processor's execution pipeline with the memory location, wherein the reconciliation ensures that the second  
25 instruction will be fetched and executed from the memory location if the program subsequently returns to that memory location.

17. The system according to claim 13, wherein the patch  
30 class instructions include no-operation instructions and branch instructions.

099389103401  
101202181660

Docket No. AUS920010676US1

18. The system according to claim 13, wherein the components are part of a multiprocessor computer system.

19. The system according to claim 13, wherein the  
5 components are part of a uniprocessor computer system.

09188T660  
40420